

## **A Computationally Efficient Path Following Control Strategy of Autonomous Electric Vehicles with Yaw Motion Stabilization**

GUO, Ningyuan, ZHANG, Xudong, ZOU, Yuan, LENZO, Basilio  
<<http://orcid.org/0000-0002-8520-7953>> and ZHANG, Tao

Available from Sheffield Hallam University Research Archive (SHURA) at:

<http://shura.shu.ac.uk/26374/>

---

This document is the author deposited version. You are advised to consult the publisher's version if you wish to cite from it.

### **Published version**

GUO, Ningyuan, ZHANG, Xudong, ZOU, Yuan, LENZO, Basilio and ZHANG, Tao (2020). A Computationally Efficient Path Following Control Strategy of Autonomous Electric Vehicles with Yaw Motion Stabilization. IEEE Transactions on Transportation Electrification, p. 1.

---

### **Copyright and re-use policy**

See <http://shura.shu.ac.uk/information.html>

# A Computationally Efficient Path Following Control Strategy of Autonomous Electric Vehicles with Yaw Motion Stabilization

Ningyuan Guo, Xudong Zhang, *Member, IEEE*, Yuan Zou, *Senior Member, IEEE*, Basilio Lenzo, *Member, IEEE*, and Tao Zhang

**Abstract**— This paper proposes a computationally efficient path following control strategy of autonomous electric vehicles (AEVs) with yaw motion stabilization. First, the nonlinear control-oriented model including path following model, single track vehicle model, and Magic Formula tire model, are constructed. To handle the stability constraints with ease, the nonlinear model predictive control (NMPC) technique is applied for path following issue. Here NMPC control problem is reasonably established with the constraints of vehicle sideslip angle, yaw rate, steering angle, lateral position error, and Lyapunov stability. To mitigate the online calculation burden, the continuation/ generalized minimal residual (C/GMRES) algorithm is adopted. The deadzone penalty functions are employed for handling the inequality constraints and holding the smoothness of solution. Moreover, the varying predictive duration is utilized in this paper so as to fast gain the good initial solution by numerical algorithm. Finally, the simulation validations are carried out, which yields that the proposed strategy can achieve desirable path following and vehicle stability efficacy, while greatly reducing the computational burden compared with the NMPC controllers by active set algorithm or interior point algorithm.

**Index Terms**— Continuation/ generalized minimal residual algorithm, fast initial solution calculation, nonlinear model predictive control, path following, yaw stability.

## I. INTRODUCTION

MODERN transportation and advanced vehicular technologies have been unconsciously improving people's lives, which also brings the higher-level requirements for autonomous vehicle (AV) control. Increasingly accurate and effective path programming technologies with continuously changing traffic environments have urgently propelled AVs to further improve its path-following effects, such as reliability, availability, safety and so forth [1]. Moreover, among all the driveline configurations, electric vehicles (EV) are the promising one and considered as the most appropriate chassis for AVs' application [2]. Compared with the internal combustion engine vehicles, EVs exhibit the outstanding advantages on environmental friendliness, high and smooth power electric supply for autonomous devices, and fast dynamics responses of motors.

Therefore, it is crucial and urgent to develop the advanced path following strategy for autonomous electric vehicles (AEV).

The primary target of path following is to reasonably manage the vehicle motion for accurately tracking a reference path and guaranteeing vehicle dynamic stability [3]. This is challenging since its control effects are sensitive to the vehicle lateral maneuverability paid by the high nonlinearity of tires [4]. Several control strategies have been proposed in the last decades. In [5], a nested proportional integral (PI) controller is designed based on vision devices in the case of roads with an uncertain curvature. In [6], a path following and lateral stability control method is proposed for a four wheels' steering AEV, where the Hamilton energy function based controller is formulated and applied for control command optimization. Ref. [7] presents an exponential-like-sliding-mode fuzzy type-2 neural network approach method for path following, which can hold the stability of the closed-loop system and adaptively adjust the sliding surface for smooth convergence of errors. In [8], an nonlinear controller is designed for path following of AEVs, which combines the advantages of composite nonlinear feedback control in improving the transient performance and integral sliding mode control in guaranteeing expected robustness.

Recently, the studies on AEVs path following try to address more practical issue [9]. The system constraints, like vehicle stability limits, safe driving area, actuator limits, are inevitable in real-world AEVs' application. Hence, it is critical and desirable to design a path following controller that considers these constraints. Moreover, it is noteworthy that the control robustness and process effects are generally two conflicted indexes that require to be traded off. Some robust controllers may be relatively conservative due to their priority of how to handle the parameter uncertainties and/or disturbances for norm-boundary limiting [10]. Under normal cycle, the lateral vehicle dynamics are closely linear, and the good controller robustness may not be a great benefit but possibly entailing the unsatisfactory control effects.

Model predictive control (MPC) technique is a promising candidate to handle the above problem. Different from other methods, the future system states are obtained per sample

This work is supported by the National Natural Science Foundation of China (Grant No. 51805030), in part by the National Natural Science Foundation of China (Grant No. 51775039), and in part by Graduate Technological Innovation Project of Beijing Institute of Technology (Grant No. 2019CX20020).

Ningyuan Guo, Xudong Zhang, Yuan Zou, and Tao Zhang are with the Beijing Collaborative and Innovative Center for Electric Vehicle and School of Mechanical Engineering, Beijing Institute of Technology, Beijing, China

(email: gny123@qq.com; xudong.zhang@bit.edu.cn; zouyuanbit@vip.163.com; ztao1208@126.com) (Corresponding author: Xudong Zhang and Yuan Zou).

Basilio Lenzo is with the Department of Engineering and Mathematics, Sheffield Hallam University, Sheffield S1 1WB, UK (email: basilio.lenzo@shu.ac.uk)

instant by updates of control-oriented model and optimized by numerical algorithms through the predictive horizon, and meanwhile the constraints can be transferred to be explicit with ease [11]. Focusing on the high nonlinearity of tires in path following, the nonlinear MPC (NMPC) is more preferable than linear MPC, while entailing greater computational burden and hindering its wider application. Accordingly, various operation forms or efficient solving algorithms are proposed. The linear time varying MPC (LTV-MPC) controllers are presented in Refs. [12, 13]. To reduce the computing labor, nonlinear model is linearized at initial states by Taylor expansion, but suffering from the effects deterioration due to model approximations [14]. In [15], focusing on lower vehicle velocity, a customized genetic algorithm achieves the real-time optimization of NMPC controller for path following. Nevertheless, only the vehicle kinematic motion is considered; that is, the vehicle lateral dynamic is not included. With the aid of parallel calculation advantages of particle swarm optimization algorithm and field programmable gate array (FPGA) chip, a real-time NMPC strategy of AEVs is developed and verified under hardware-in-the-loop test [16]. However, the FPGA chip is difficult for large-scale application due to its high expense. Explicit MPCs (EMPC) are also a computational efficient approach for real-time implementation, whose control problem is optimized offline to generate the look-up tables adopted online. Ref. [17] points out that EMPC yields the similar effects to nominal NMPC. That said, such performance relies on a mass of points defined in look-up tables, resulting in high memory requirements that limit its applicability.

At the startup time point of path following, the initial solution selection is also of importance for control effects in MPCs. Indeed, the zero control command (i.e., initial solution in MPC optimization) is the optimal solution under a case that the initial position and heading angle errors are zero. Nevertheless, the initial path following errors are possible nonzero in practice. The zero initial solution or one by trial and error in MPC optimization may deteriorate the solving optimality, cause the solving divergence, and even lead to the loss of vehicle stability. Increasing the maximum iteration numbers and/or the iteration toleration error by numerical algorithms is an effective approach to gain the good initial solution, but entailing the huge calculation time.

To fill up above gaps, a computationally efficient NMPC control strategy for path following of AEVs is proposed in this paper. To meet the high nonlinearities of vehicle, the control-oriented model is constructed by integrating the path following model, single track vehicle model, and the Magic Formula (MF) tire model. The NMPC problem is established for minimizing the lateral position and heading angle errors, and meanwhile the constraints regarding sideslip angle, yaw rate, front wheels' steering angle, lateral position, and Lyapunov stability, are imposed. For real-time optimization in NMPC, the continuation/ generalized minimal residual (C/GMRES) algorithm is applied. The deadzone penalty functions are also employed in the C/GMRES algorithm for inequality constraints handling while achieving the solution's smoothness. A varying predictive duration method is introduced so that the good initial solution can be fast gained by numerical iterative algorithms. Finally, the simulations comprehensively verify the effectiveness of proposed strategy in path following and vehicle

stability control, which does not only yield the superior control efficacy but the desirable computational efficiency.

The remainder of this paper is organized as follows. Section II introduces the control-oriented model in strategy design, and the control strategy is illustrated in Section III with details. Section IV shows and analyzes the strategy validation results, followed by the key conclusions in Section V.

## II. SYSTEM MODELLING FOR PATH FOLLOWING

The studied AEV is a passenger car with two axles and four wheels, and the objective in this paper is to conduct the desired tracking performance for target path with guaranteed lateral vehicle stability. In the following, the path following model and yaw motion model are illustrated in orders first, and then the control-oriented model is established.

### A. Path Following Model

Defining the lateral distance between vehicle's position and desired path as  $y_e$  and the heading angle error to the path centerline as  $\psi_e$ , their derivation can be expressed as below [18],

$$\begin{cases} \dot{y}_e = \beta v_x \cos \psi_e + v_x \sin \psi_e \\ \dot{\psi}_e = \dot{\psi} - \dot{\psi}_r = \gamma - \dot{\psi}_r \end{cases} \quad (1)$$

where  $v_y$ ,  $v_x$ ,  $\psi$ ,  $\beta$ ,  $\gamma$  and  $\psi_r$  are the vehicle lateral and longitudinal velocity, vehicle heading angle, vehicle sideslip angle, vehicle yaw rate, as well as the reference heading angle of desired path, respectively. We assume that the vehicle sideslip angle can be accurately estimated by advanced algorithms [19] and limited near zero by the proposed controller, thus  $\tan \beta \approx \beta$  [20] is adopted in Eq. (1).

To gain  $\dot{\psi}_r$ , another assumption is made that the vehicle proceeds with constant longitudinal velocity in the predictive horizon. Defining the state update time step in controller as  $\Delta \tau$ , the preview distances can be calculated as  $j v_x \Delta \tau$ ,  $j=0,1,\dots,n$ , and the corresponding positions of preview road points in global coordinate can be measured by sensors and represented as  $[X_j, Y_j]$ ,  $j=0,1,\dots,n$ , as shown in Fig. 1 (a). Then, the collected preview road points are fitted by quintic polynomial expression as  $Y_j = \chi(X_j)$ , and  $\dot{\psi}_r$  in the preview sight is presented as,

$$\dot{\psi}_r = v_x \frac{\chi''}{1 + \chi'^2} \quad (2)$$

Importing preview road point  $X_j$  and  $v_x$  into Eq. (2),  $\dot{\psi}_r$  can be gained and adopted in the controller for state updates.

*Remark 1:* Unlike the general method that a fixed preview road point is applied as the control reference in path following model [21], the tracking reference here is actually a sequence related to state update time points for more desirable model accuracy of Eq. (1).

### B. Yaw Motion Model of Vehicle

The single track vehicle model is applied here to characterize the vehicle lateral dynamic, as shown in Fig. 1 (b), whose expression can be furnished as,

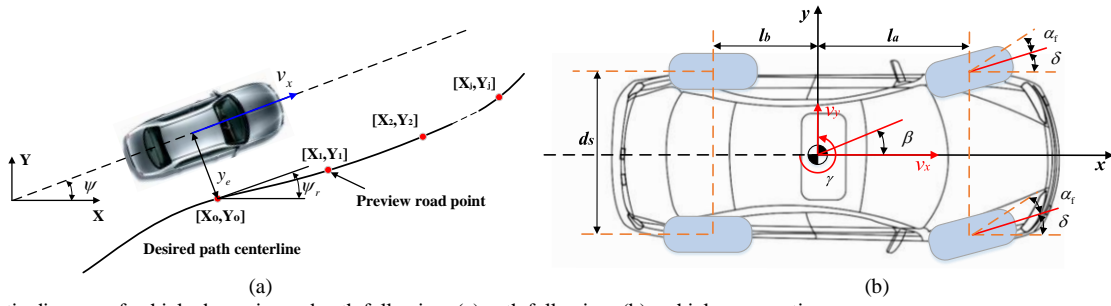


Fig. 1. Schematic diagram of vehicle dynamics and path following. (a) path following; (b). vehicle yaw motion.

$$\begin{cases} m_v(\dot{v}_y + v_x\gamma) = F_{yf} + F_{yr} \\ I_z\dot{\gamma} = l_a F_{yf} - l_b F_{yr} \end{cases} \quad (3)$$

where  $m_v$ ,  $l_a$ ,  $l_b$ , and  $I_z$  index the total vehicle mass, the distance from front axle to the center of gravity (CG), the distance from rear axle to CG, the vehicle yaw moment, and yaw moment inertia, respectively.  $v_x$  and  $v_y$  represent the longitudinal and lateral velocity for vehicle's CG, respectively.  $F_{yf}$  and  $F_{yr}$  respectively denote the lateral forces of front and rear tires, which are expressed as,

$$\begin{cases} F_{yf} = F_{yfl} + F_{yfr} \\ F_{yr} = F_{yrl} + F_{yrr} \end{cases} \quad (4)$$

where the subscripts of "fl", "fr", "rl", and "rr" express that the corresponding variables are related to front, rear, front-left, front-right, rear-left and rear-right wheels, respectively and hereinafter. The tire sideslip angles of front and rear wheels, i.e.,  $\alpha_f$  and  $\alpha_r$ , can be calculated by,

$$\begin{cases} \alpha_f = \delta - (v_y + l_a\gamma) / v_x = \delta - (\beta + l_a\gamma / v_x) \\ \alpha_r = (l_b\gamma - v_y) / v_x = l_b\gamma / v_x - \beta \end{cases} \quad (5)$$

where  $\delta$  is the steering angle of front wheels. The MF tire model is employed to determine the highly nonlinear tire features under pure slip cases, whose expression is furnished as, [22],

$$\begin{aligned} F_y &= MF(\alpha, \mu, F_z) \\ &= D_o \sin \left\{ C_o \arctan \left[ B_o \alpha - E_o (B_o \alpha - \arctan(B_o \alpha)) \right] \right\} \end{aligned} \quad (6)$$

$$\begin{cases} B_o = C_{Fa} / (C_o D_o) \\ C_{Fa} = B_o C_o D_o = c_1 \sin(2 \arctan(F_z / c_2)) \\ D_o = \mu F_z \end{cases} \quad (7)$$

where  $\mu$  and  $F_z$  denote the road adhesion coefficient and the vertical load of the tire, respectively.  $B_o$  is the stiffness factor,  $C_{Fa} = B_o C_o D_o$  is the cornering stiffness, and  $D_o$  is the peak factor. The shape factors  $C_o$ ,  $E_o$ , and the parameters  $c_1$  and  $c_2$  are determined through least-squares approximation [23]. By Eqs. (3) to (7), the yaw motion of vehicle can be presented:

$$\begin{cases} \dot{\beta} = 2(MF(\alpha_f, \mu, F_{zf}) + MF(\alpha_r, \mu, F_{zr})) / m_v v_x - \gamma \\ \dot{\gamma} = 2(l_a MF(\alpha_f, \mu, F_{zf}) - l_b MF(\alpha_r, \mu, F_{zr})) / I_z \end{cases} \quad (8)$$

### C. System Model Construction for Controller Design

According to Eqs. (1) and (8), the path following system model can be yielded as,

$$\begin{cases} \dot{x} = f(x, u, w) \\ y = C_1 x \\ \xi = C_2 x \end{cases} \quad (9)$$

and

$$C_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, C_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

where  $x = [\beta \ \gamma \ y_e \ \psi_e]^T$  is the system state variable, and  $u = \delta$  is the system control variable.  $w = \dot{\psi}_r$  is arranged as the external disturbance.  $y$  is the measured output, and  $\xi = [y_e \ \psi_e]^T$  is the controlled output. One should note that for simplification, the dynamic responses of steering motor and traction motor are neglected in the control-oriented model but imposed in the validation model, which will be described in Section IV. Now the formulation of system model is completed, and the proposed control strategy will be illustrated in the following.

### III. PATH FOLLOWING CONTROL STRATEGY

In this section, a computationally efficient path following control strategy is introduced. First, the NMPC control problem is concisely built, and then the C/GMRES algorithm is described, followed by the handling of inequality constraints and the varying predictive duration for fast initial solution optimization.

#### A. Nonlinear Model Predictive Control Problem

The NMPC control problem is constructed as,

$$\begin{aligned} \min \quad & J_{nmpc} = \int_{t_o}^{t_o + N_p - 1} l(\xi(\tau), u(\tau)) d\tau \\ \text{s.t.} \quad & \dot{x}(\tau) = f(x(\tau), u(\tau)) \\ & x_o = x(t_o) \\ & h(x(\tau), u(\tau)) \leq 0 \end{aligned} \quad (11)$$

where  $x_o$  is the initial system state.  $N_p$  represents the steps of predictive horizon and equals to the control horizon in this paper.  $t_o$  is the current sample instant of controller.

$l(\xi(\tau), u(\tau))$  is the performance cost in the form of least squares norm:

$$l(\xi(\tau), u(\tau)) = (\xi(\tau) - \xi_r)^T Q (\xi(\tau) - \xi_r) + u(\tau)^T R u(\tau) \quad (12)$$

where  $Q = \text{diag}\{q_1 \ q_2\}$  and  $R$  express the weight factors corresponding to controlled output error and control variable, respectively.  $\xi_r = [0 \ 0]^T$  is the reference output, and the inequality constraints  $h(x(\tau), u(\tau))$  are set as,

$$\begin{cases} \beta_{\min} \leq \beta \leq \beta_{\max} \\ \gamma_{\min} \leq \gamma \leq \gamma_{\max} \\ \delta_{\min} \leq \delta \leq \delta_{\max} \\ y_{e\min} \leq y_e \leq y_{e\max} \end{cases} \quad (13)$$

$$\frac{\partial V}{\partial x} f(x(t_o), u(t_o)) \leq \frac{\partial V}{\partial x} f(x(t_o), \gamma_{aux}(t_o)) \quad (14)$$

where the subscripts of “max” and “min” mean the allowable maximum and minimum related variables, respectively. The boundaries of  $\beta$  and  $\gamma$  in Eq. (13) are defined as

$$\begin{cases} \beta_{\max} = -\beta_{\min} = \arctan(0.02\mu g) \\ \gamma_{\max} = -\gamma_{\min} = \mu g / v_x \end{cases} \quad [24], \text{ where } g \text{ is the}$$

gravitational acceleration. To guarantee the drive safety of vehicle, the restriction of  $y_e$ , i.e., the boundaries  $y_{e\min} = -(d_r - d_s)/2$  and  $y_{e\max} = (d_r - d_s)/2$  [25], are set up as the constraint in Eq. (13). Here  $d_r$  is the single road width, and  $d_s$  is the vehicle track width. Moreover, to hold the closed-loop stability of the NMPC controller, an additional constraint regarding the Lyapunov function  $V(\cdot)$  and an auxiliary control law is adopted and listed in Eq. (14), where  $\gamma_{aux}$  is the auxiliary control law of vehicle yaw rate and can be set up by any Lyapunov-based control method. For simplification, a nonlinear control law by backstepping method is employed here [26],

$$\gamma_{aux} = \dot{\psi}_r - k_2 k_1 (y_e + \frac{1}{k_1} \psi_e) \quad (15)$$

where  $k_1$  and  $k_2$  are two adjustment factors, and  $k_2 > v_x k_1$  is a necessary condition to simultaneously make  $y_e \rightarrow 0$  and  $\psi_e \rightarrow 0$ . From [26], the corresponding Lyapunov function of Eq. (15) is expressed as,

$$V = \frac{1}{2} y_e^2 + \frac{1}{2k_1^2} (\psi_e + k_1 y_e)^2 \quad (16)$$

**Theorem 1:** Assuming that the target reference is bounded and smooth, and the recursive feasibility is hold, the proposed close-loop system with constraint (14) is asymptotically stable regarding the path following equilibrium  $[y_e, \psi_e]^T = [0, 0]^T$ . That is, the AEV will converge to the desired path by constructed NMPC controller.

**Proof [27]:** Since the Lyapunov function Eq. (16) is continuously differentiable and radically unbounded, according to the converse Lyapunov theorems, there exist a group of class  $K_\infty$  functions  $\varpi_i(\cdot)$ ,  $i=1, 2, 3$ , that make the following expression hold:

$$\begin{cases} \varpi_1(\|x\|) \leq V(x) \leq \varpi_2(\|x\|) \\ \frac{\partial V}{\partial x} f(x, \gamma_{aux}(x)) \leq -\varpi_3(\|x\|) \end{cases} \quad (17)$$

Because  $\gamma_{aux}(x)$  only contributes to the first state update calculation, combining with Eq. (14), we have

$$\frac{\partial V}{\partial x} f(x, u) \leq \frac{\partial V}{\partial x} f(x, \gamma_{aux}(x)) \leq -\varpi_3(\|x\|) \quad (18)$$

By Lyapunov stability theorem [28], the closed-loop system is asymptotically stable.

### B. C/GMRES Algorithm and Its Application

Given the high nonlinearities regarding system model (9) and control problem, the calculation burden in NMPC is extremely large, causing the difficulty of real-time optimization. Hence, the C/GMRES algorithm is proposed in this paper. It is a combined algorithm by integrating the continuation method and the GMRES algorithm, whose calculation process is explicit so that the number of mathematical operations at each sample is fixed ensuring the finite computational time [29]. Moreover, its derivation is related to the globally optimality conditions such that the optimization quality can be guaranteed. Thus, it is a well-suited approach for the addressed issue.

#### 1) C/GMRES Algorithm

Taking one-dimensions control variable to illustrate, a general NMPC control problem can be shown as,

$$\begin{aligned} \min \quad & J = g(x(t_o + N_p), u(t_o + N_p)) + \int_{t_o}^{t_o + N_p - 1} l(x(\tau), u(\tau)) \\ \text{s.t.} \quad & \dot{x}(\tau) = f(x(\tau), u(\tau)) \\ & x_o = x(t_o) \\ & C(x(\tau), u(\tau)) = 0 \end{aligned} \quad (19)$$

where  $C(x(\tau), u(\tau))$  means the equality constraints, and  $g(x(t_o + N_p), u(t_o + N_p))$  is the terminal cost. Based on Pontryagin's minimum principle (PMP) [30], the Hamiltonian function of Eq. (19) can be furnished,

$$H(x(\tau), u(\tau), \lambda^T(\tau)) = l(x(\tau), u(\tau)) + \lambda^T(\tau) \times f(x(\tau), u(\tau)) + \mathcal{G}^T C(x(\tau), u(\tau)) \quad (20)$$

where  $\lambda \in \mathbb{R}^{n_x}$  represents the co-state vector, and  $\mathcal{G} \in \mathbb{R}^{n_c}$  denotes the Lagrange multipliers associated with equality constraints.  $n_x$  and  $n_c$  denote the dimensions of state variables and equality constraints, respectively. The necessary conditions to find the optimal solution can be described by PMP as,

$$u^*(\tau) = \arg \min [H(x^*(\tau), u(\tau), \lambda^{*T}(\tau))] \quad (21)$$

$$x^*(\tau + 1) = f(x^*(\tau), u^*(\tau)) \Delta \tau + x^*(\tau) \quad (22)$$

$$\lambda^*(\tau) = \lambda^*(\tau + 1) + \frac{\partial H}{\partial x} (x^*(\tau + 1), u(\tau + 1), \lambda^{*T}(\tau + 1)) \Delta \tau \quad (23)$$

$$x_o(t) = x(t_o) \quad (24)$$

$$\lambda(t_o + N_p - 1) = \frac{\partial g}{\partial x} (x^*(t_o + N_p - 1)) \quad (25)$$

$$C(x(\tau), u(\tau)) = 0 \quad (26)$$

The optimized vector can be set as  $U(t_o) = [u(t_o)^T \ \cdots \ u(t_o + N_p - 1)^T \ \mathcal{G}(t_o)^T \ \cdots \ \mathcal{G}(t_o + N_p - 1)^T]^T$ .  
(1+n<sub>c</sub>)N<sub>p</sub><sup>T</sup>

According to Eqs. (20) to (26), the optimization problem can be reformulated by recursive calculations:

$$F(U(t_o), x_o(t)) = \begin{bmatrix} \frac{\partial H}{\partial u}(x^*(t_o), u^*(t_o), \lambda^{*T}(t_o)) \\ \vdots \\ \frac{\partial H}{\partial u}(x^*(t_o + N_p - 1), u^*(t_o + N_p - 1), \lambda^{*T}(t_o + N_p - 1)) \\ C(x^*(t_o), u^*(t_o), \lambda^{*T}(t_o)) \\ \vdots \\ C(x^*(t_o + N_p - 1), u^*(t_o + N_p - 1), \lambda^{*T}(t_o + N_p - 1)) \end{bmatrix} = 0 \quad (27)$$

Undoubtedly, Eq. (27) can be solved by the numerical iteration algorithms, like trust-region-dogleg (TRD) and interior point (IP) methods, while making the computationally expensive and inefficient. To avoid the calculations in Eq. (27) regarding Jacobian matrix, Hessian matrix and inverse, the C/GMRES algorithm is introduced. Based on continuation method [31],  $F(U(t_o), x_o(t), t)$  can be transformed as a linear dynamic system,  $\dot{F}(U(t_o), x(t), t) = -\psi_s F(U(t_o), x(t), t)$ , where  $\psi_s$  is the stability matrix for stabilizing  $F(U(t_o), x_o(t), t)$  at original. Assuming  $\frac{\partial F}{\partial U}(U(t_o), x(t_o), t)$  is nonsingular, the solution  $\dot{U}(t_o)$  is decided:

$$\dot{U}(t_o) = \left[ \frac{\partial F}{\partial U}(U(t_o), x(t), t) \right]^{-1} [-\psi_s F(U(t_o), x(t), t) - \frac{\partial F}{\partial x}(U(t_o), x(t), t) \times \dot{x}(t)] \quad (28)$$

To reduce the calculation time caused by the Jacobians of  $F$  and the inversion of  $\frac{\partial F}{\partial U}$ , the forward difference approximation is adopted, and Eq. (28) can be rewritten as,

$$D_h F(U(t_o), x(t) + \dot{x}h, t + h : \dot{U}, 0, 0) = -\psi_s F(U(t_o), x(t), t) - D_h F(U(t_o), x(t), t : 0, \dot{x}, 1) \quad (29)$$

where  $h$  is a positive real value. Now the original problem is approximately transformed as a linear equation regarding  $\dot{U}$  and can be efficiently solved by GMRES algorithm [32]. The GMRES algorithm is one of the Krylov-subspace methods that is suitable to solve the large sparse linear equations for minimization of residual. The advantage of this algorithm is that, in principle, it can reduce the residual monotonically and converges the optimal solution within the same iterations as the dimension of the given equation [33]. After obtaining the optimal  $\dot{U}$ , the control command  $U$  can be calculated by tracing  $\dot{U}$  regarding the sample step. The “warm-startup” mechanism is arranged here to improve the convergence rate of algorithm [34]. To summarize, the overall calculation steps of C/GMRES algorithm are illustrated in Table I, and the detailed error analysis of C/GMRES algorithm can be found in [35].

## 2) Handling Inequality Constraints

The traditional C/GMRES algorithm is unable to tackle the inequality constraints in optimization. Moreover, the fast numerical convergence of optimal solution is yielded only

when the expected solution smoothness is given [33]. Hence, the deadzero-quadratic penalty function from softplus rectifier

TABLE I. C/GMRES ALGORITHM CALCULATION STEPS.

Algorithm 1	C/GMRES algorithm
<b>Step 1:</b>	Set $t = 0$ . Gain the initial state variable $x_o = x(0)$ and find $U(0)$ by analytic or numerical methods to make $\ F(U(0), x_o(0))\  \leq \kappa$ , where $\kappa$ is a small positive constant.
<b>Step 2:</b>	For $\tilde{t} \in [t_o, t_o + N_p]$ , set $u(t_o)$ equals to the first element in $U(\tilde{t})$ and insert into plant.
<b>Step 3:</b>	For $\tilde{t} \in [t_o, t_o + N_p]$ , set $U(\tilde{t}) = U(\tilde{t} - \Delta T)$ as the initial solution, where $\Delta T$ represents the sample instant. The operation $U(\tilde{t}) = U(\tilde{t} - \Delta T)$ is the warm start mechanism to speed up optimization convergence. Then, gain the initial state variable $x_o = x(t_o)$ and compute $\dot{U}(\tilde{t})$ by Eq. (29) and GMRES algorithm.
<b>Step 4:</b>	Calculate $U(\tilde{t}) = U(\tilde{t} - \Delta T) + \dot{U}(\tilde{t})\Delta T$ , and go back to <b>Step 2</b> .

is adopted in this paper [36]:

$$\phi(z) = \left( \ln(1 + e^{z - z_{\max}}) + \ln(1 + e^{-z + z_{\min}}) \right)^2 \quad (30)$$

where  $z$  represents a variable required to be limited. The main advantage of Eq. (30) is its convex feature so as to efficient computation and gradient propagation [37]. Now the control objective in Eq. (11) can be transformed as below,

$$\begin{aligned} \min \quad & J_{nmpc} = \int_{t_o}^{t_o + N_p - 1} l(\xi(\tau), u(\tau)) + \sum_j \rho_j \phi_j(x(\tau), u(\tau)) d\tau \\ \text{s.t.} \quad & \dot{x}(\tau) = f(x(\tau), u(\tau)) \\ & x_o = x(t_o) \end{aligned} \quad (31)$$

where  $\rho_j$  is the  $j$ th weight coefficient. By Eq. (31), C/GMRES algorithm can be applied to obtain the expected control command.

**Remark 2:** To handle the inequality constraints, there are the auxiliary variable method, the barrier function method, and the external penalty method. For the auxiliary variable method, the inequality constraints are transformed to be a group of same-dimensional equality constraints, where the dummy variables  $\varsigma_j$  are adopted to make  $h_j(x(t_o), u(\tau)) + \varsigma_j^2 = 0$  [35]. Meanwhile, to avoid the singularity points in optimization, a cost item of  $\varsigma_j$  is added into performance index, like

$$\tilde{l}(x(\tau), u(\tau)) = l(x(\tau), u(\tau)) - \sum_j \rho_j \varsigma_j, \text{ where } \tilde{l}(x(\tau), u(\tau)) \text{ is}$$

defined as the cost function in performance index, and  $\rho_j$  is the weight coefficient of the  $j$ th penalty item. Nevertheless, this method is proven to be hard for stabilization and parameters' tuning [38]. For the barrier function method, a additional cost item of the log function is constructed in the performance according to the inequality constraints [39]. This method is validated to have superior convergence and widely used as a analytic approach for handling inequality constraints in various algorithms. But, it is only effective when the optimization starts from the system control commands and the feedback states that are within constraints, else it will cause

reverse penalty. Since inevitable errors exist between vehicle plant and the control-oriented model, the barrier function is unsuitable for this pathing following issue, especially under extreme drive cycle. For the external penalty method, the additional cost item can be expressed as

$$\varsigma_j(x(\tau), u(\tau)) = \begin{cases} 0, & h_j(x(\tau), u(\tau)) \leq 0 \\ \rho_j \times h_j(x(\tau), u(\tau))^2, & h_j(x(\tau), u(\tau)) > 0 \end{cases} \quad \text{in}$$

performance index to avoid out of the boundary. It is known to be valid and easily tuned, while the differentiability and continuity are not so ideal leading to the deterioration of solving quality and convergence to some extent [36].

More intuitively, Fig. 2 shows an illustration for the symmetrical boundary regarding  $z_{\max} = z_{\min} = 10$  by above three functions. Although the interval of  $z$  is selected to be 0.0001, the value of log barrier function (i.e., barrier function method) at  $z = -10$  cannot be plotted by numerical values, leading to the ill-conditioned when adopted to handle the inequality constraints. Moreover, due to model errors existing in the addressed path following issue, the feedback states may be near but out of the boundaries, which by barrier function method will cause the reverse penalty of state variables to further keep away from the allowable range. Although the external penalty function does not lead to the above problems, its differentiability and continuity at boundaries  $z_{\max} = z_{\min} = 10$  are worse than the deadzone penalty function. Therefore, the deadzone penalty functions are selected for inequality constraints handling in this paper.

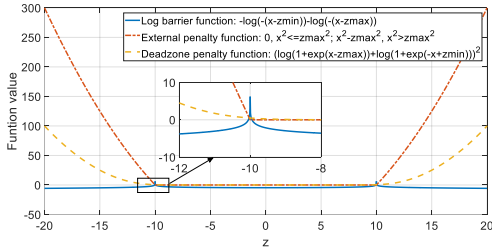


Fig. 2. An illustration of the deadzone penalty function ( $z_{\max} = z_{\min} = 10$ ).

### 3) Varying Predictive Duration for Fast Gaining Initial Solution

When there exist initial state errors, such as the lateral position and heading angle errors, the initial solution quality is sensitive to optimization convergence in NMPC issue. To fast initial solution optimization, the varying predictive duration is imposed in this paper, and the time-dependent duration of predictive horizon is furnished as [35],

$$T(t) = T_f \times (1 - e^{-\varepsilon t}) \quad (32)$$

where  $T(t)$  is the duration of predictive horizon;  $T_f$  is a given time duration constant;  $t$  is a time value that records the time duration of controller operation, which is reset to zero when the controller restarts; and  $\varepsilon$  is a coefficient and determines the increase rate of  $T(t)$ . With the greater  $\varepsilon$ ,  $T(t)$  increases more rapidly and finally infinitely inclines to  $T_f$  as  $t \rightarrow \infty$ . By Eq. (32), the state update step  $\Delta\tau$  in controller and the preview road points are determined by the following expression:

$$\Delta\tau = T(t) / N_p \quad (33)$$

By Eqs. (32) and (33), the predictive duration  $T(t)$  is near zeros at the startup instant of controller, thus the initial solution of  $N_p$  variables in optimization can be considered to be only one. By this manner, at the initial period, the numerical algorithms, like IP and active set (AS) algorithms, can be employed for initial solution optimization in a computational efficient way.

*Remark 3:* By varying predictive duration, the initial solution of one element is optimized by numerical algorithms only at the time point of controller startup. At a generic instant,  $N_p$  optimized variables are defined and gained by the C/GMRES algorithm under the “warm-startup” mechanism. During the beginning period, predictive horizon length by varying predictive duration is constantly changing, but the NMPC optimization problem is essentially same with that of constant predictive duration. Hence the varying predictive duration is applicable to the NMPC. Moreover, whether C/GMRES algorithm successfully optimizes and guarantees the convergence rate or not is determined by the continuity and smoothness of optimization problem rather than by the predictive duration. To sum up, the varying predictive duration method is available to the proposed C/GMRES algorithm based NMPC controller.

*Remark 4:* At the beginning period of controller startup, the short predictive duration may worsen the control effects to some extent. However, the total predictive duration  $T_f$  is generally relatively short in path following focusing on the fast dynamics response of AEVs, and one can also raise the increase rate of predictive length by adjusting coefficient  $\varepsilon$  to reduce the adverse influences of short predictive size. Therefore, it is acceptable by the varying predictive duration method to fast gain the good initial solution. The validation and analysis regarding the control effects by this method will be illustrated in the next section.

## IV. NUMERICAL SIMULATION, VALIDATIONS, AND ANALYSIS

In this paper, the co-simulation platform combining by software Matlab/Simulink<sup>®</sup> and Carsim<sup>®</sup> are adopted for effectiveness validation of the proposed strategy under Intel(R) Core(TM) i7-7700HQ CPU @ 2.8GHz laptop computer, where the control strategy and an embedded vehicle model are constructed by Matlab/Simulink<sup>®</sup> and Carsim<sup>®</sup>, respectively. The embedded vehicle model in CarSim<sup>®</sup> consists of steering mechanic, suspension components, tires, and so forth, which is extensively verified and correlated to reproduce the real-world vehicle performance according to the measured data by many automotive manufacturers [40]. Therefore, it is believed with high confidence and accuracy in literature and industry for the studies of vehicle dynamics and control strategy development [24, 41]. In this paper, a proportional-integral controller is built in Matlab/Simulink<sup>®</sup> to generate the traction torque for desirable velocity, and the first-order inertia element modules of time constant 0.1 in Matlab/Simulink<sup>®</sup> are respectively arranged before the CarSim<sup>®</sup> imports of steering angle command and traction torque to reproduce the dynamics responses of steering and traction motors. The parameters regarding vehicle and proposed strategy are listed in TABLE II. All the results are conducted under double-lane-change (DLC)



drive cycle, and its curvature and the related path profile are shown in Fig. 3.

TABLE II. PARAMETERS REGARDING VEHICLE AND PROPOSED STRATEGY.

Parameter	Value	Unit
Vehicle mass $m_v$	1412	kg
Distance from CG to front axle $l_a$	1.015	m
Distance from CG to rear axle $l_b$	1.895	m
Coefficient of MF model $c_1$	$2.664 \times 10^5$	-
Coefficient of MF model $c_2$	$3.334 \times 10^4$	-
Coefficient of MF model $C_o$	2.725	-
Coefficient of MF model $E_o$	1.198	-
Wheel track $d_s$	1.675	m
Wheel radius $r_w$	0.308	m
Vehicle rotational inertia of Z axis $I_z$	1536.7	kg.m <sup>2</sup>
Predictive horizon steps $N_p$	10	-
Sample cycle of controller $\Delta t$	0.02	s
Weight matrix of output state $Q$	$\text{diag}\{1 \times 10^4, 202.6\}$	-
Weight matrix of control increment $R$	$\text{diag}\{5.5829 \times 10^3\}$	-
Weight coefficients of penalty items $\rho_j$	[14, 340, 1900, 270, 2800]	-
Boundary of front wheels' steering angle $\delta$	[-0.7854, 0.7854]	rad
Time duration constant $T_f$	0.2	-
Coefficient regarding varying rate $\varepsilon$	10	-
Stability matrix in C/GMRES algorithm $\psi_s$	50	-
Auxiliary control law adjustment factor $k_1$	$3/v_x$	-
Auxiliary control law adjustment factor $k_2$	3.3	-
Single road width $d_r$	4	m

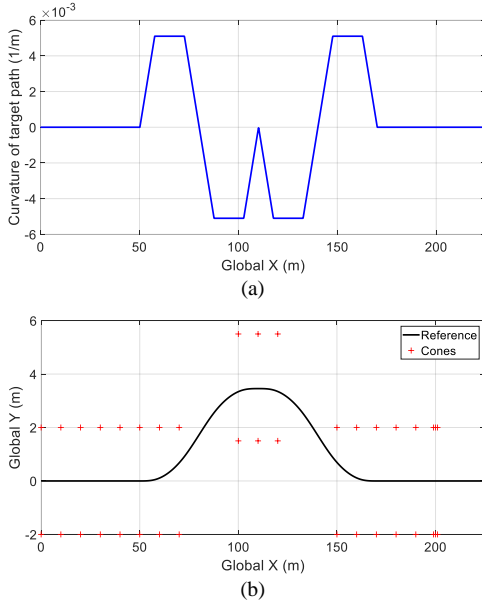


Fig. 3. DLC test cycle illustration. (a) path curvature; (b) path profile.

#### A. Validation of Path Following and Vehicle Stabilization

This validation aims to verify the path following and yaw stability effects under two extreme drive cases: Case 1 and Case 2. In Case 1, the longitudinal velocity and the road adhesion coefficient are 100 km/h and 0.85, respectively, and Case 2 is achieved under longitudinal velocity of 80 km/h and the road adhesion coefficient of 0.4. The initial position and heading angle are set as [0, 0] m and 0 deg, respectively; that is, no initial

path following errors exist in this validation. Except the proposed NMPC controller, the linear quadratic regulator (LQR) controller, the traditional NMPC controller, and the proposed NMPC without constraints, are all devoted to comprehensively illustrate the control performance by comparisons. Here the control-oriented model of LQR controller is built by Eq. (1) under the assumption of  $\cos \psi_e \approx 1$  and  $\sin \psi_e \approx \psi_e$ . Since LQR cannot directly think over the inequality constraints, its controller parameters are determined by Bryson's rule [42] in premise of satisfying of vehicle stability. The optimization in traditional NMPC controller is implemented by IP algorithm, where the Matlab<sup>®</sup> library function "fmincon" with the iteration toleration error of 0.01 is adopted. It is noteworthy that all the control problem and parameters for NMPC controllers are same except the weight coefficients of inequality constraints by deadzone penalty functions.

Fig. 4 depicts the path following errors and the vehicle steering angle command, and their corresponding vehicle yaw motion results are shown in Fig. 5. From Fig. 4, compared with LQR controller, the lateral position errors by other methods are smaller and restricted within  $\pm 0.0715$  and  $\pm 0.05$ , respectively. Moreover, the traditional NMPC and proposed NMPC controllers achieve the distinctly less heading angle errors than LQR controller. Without the consideration of constraints, the NMPC controller yields frequent fluctuations on lateral position errors, heading angle errors, and steering angle. This is explained that since only minimization of path following errors is thought over in controller, the vehicle yaw rate is outside its limits during turning, meaning that the vehicle lateral acceleration has been greater than its allowable maximum and causing the loss of vehicle stability, as shown in the subfigures of Fig. 5 (a) and (b). The undesirable stability loss leads to deletion of vehicle steering capacity and model mismatching, ultimately resulting in the swing steering effects in Fig. 4. Instead, owing to the effective constraints handling considering yaw and controller stabilities, traditional and proposed NMPC controllers yield smoother transient performance and faster convergent rate at the end of DLC cycle.

From Fig. 5, the sideslip angles by four methods are all within the limits, and except the controller without considering constraints, the others can restrict the enlargement of vehicle yaw rate and guarantee the vehicle stability. One should be noted that for LQR method, the path tracking and vehicle stability are actually two contradictory objectives, and the satisfaction of vehicle stability (i.e., the limit of yaw rate) can only be achieved by adjusting the controller parameters. Hence to more intuitively compare the path tracking effects, the control parameters are reasonably tuned in LQR controller with the preference of vehicle stability, which explains why its yaw rate is bounded in Fig. 5. In contrast, the proposed NMPC controller can carry out the expected path tracking performance in accordance with vehicle stability, as shown in Fig. 4 and Fig. 5. At around 4.5 s in Fig. 5 (a) and 5.7 s in Fig. 5 (b), interestingly, the yaw rate by Matlab<sup>®</sup> library function "fmincon" in traditional NMPC controller exceeds its limits. This is because, since there inevitably exists errors between vehicle plant and the control-oriented model, the future yaw rate trajectory is conducted to be inside constraints during



optimization but cannot be absolutely guaranteed in real values. Moreover, it can be observed that the real yaw rate of proposed strategy is bounded owing to the adopted deadzone penalty method and the well-tuned weight coefficients.

From Fig. 4 and Fig. 5, the proposed NMPC and traditional NMPC controllers yield the similar control effects no matter in path following, yaw motion, or steering angle command, indicating the near-optimality of the proposed C/GMRES algorithm. The differences between them are risen by the approximations of continuation method and forward difference method in the C/GMRES algorithm. However, it is acceptable because the near-optimal control efficacy can be implemented

while only paying distinctly smaller calculation burden than IP algorithm, as can be found in the following computational efficiency comparison of Section IV C.

To further illustrate the vehicle yaw stability, Fig. 6 depicts the phase plane regarding sideslip angle and sideslip angle rate. Under Case 1 and Case 2, the phase trajectories of the methods that considers the vehicle stability, are effectively restricted near the original point, yielding their desirable vehicle yaw stability. One can see that the NMPC controller without considering constraints shows distinctly greater moving range of phase trajectory since it only takes the path tracking into account.

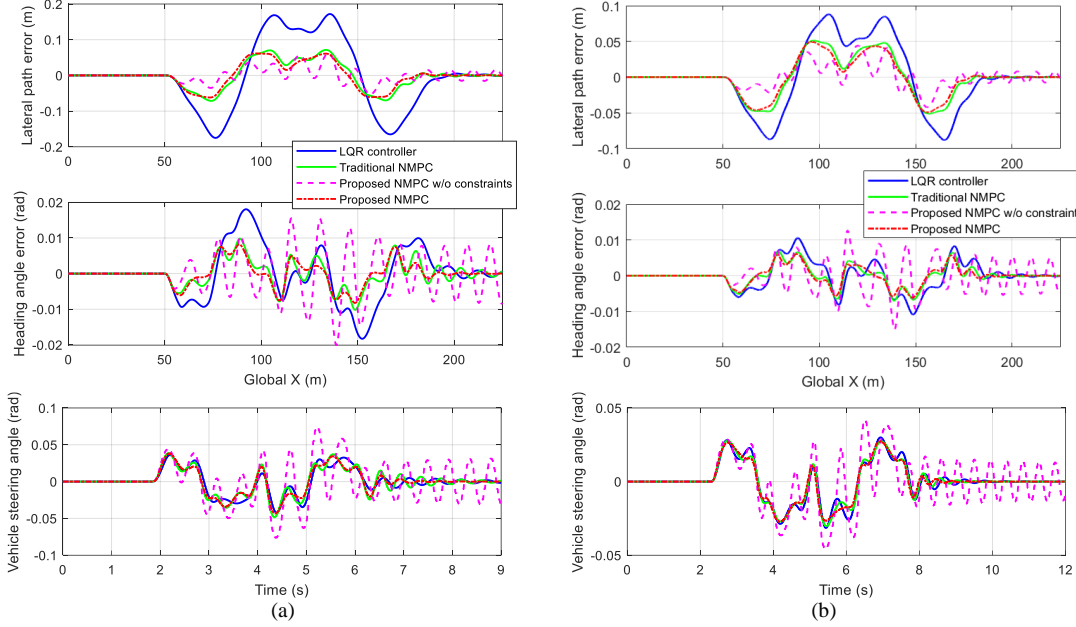


Fig. 4. Lateral position errors and heading angle errors under DLC drive cycle. (a) Case 1; (b) Case 2.

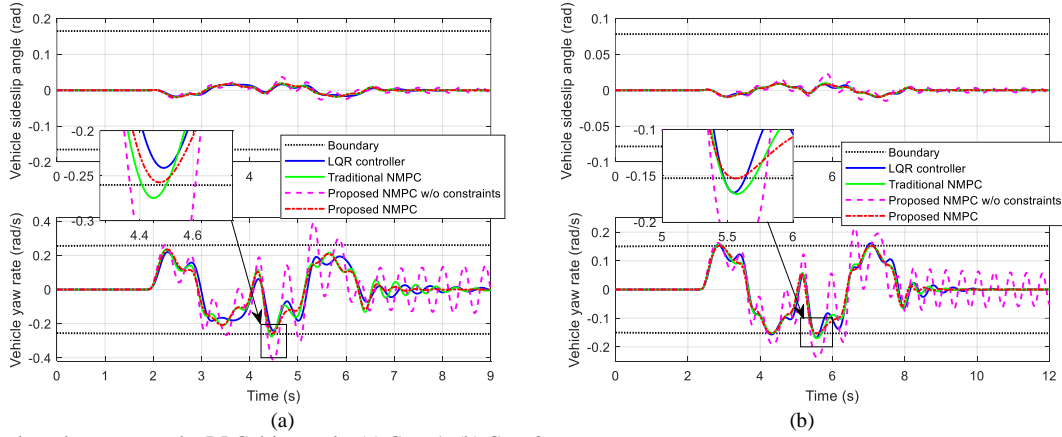


Fig. 5. Sideslip angle and yaw rate under DLC drive cycle. (a) Case 1; (b) Case 2.

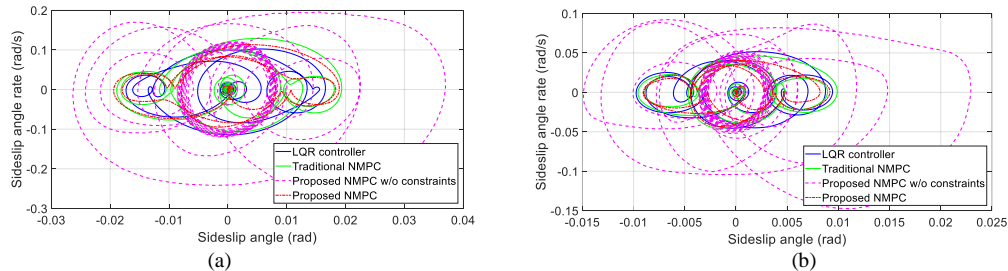


Fig. 6. Phase plane of sideslip angle and sideslip angle rate. (a) Case 1; (b) Case 2.

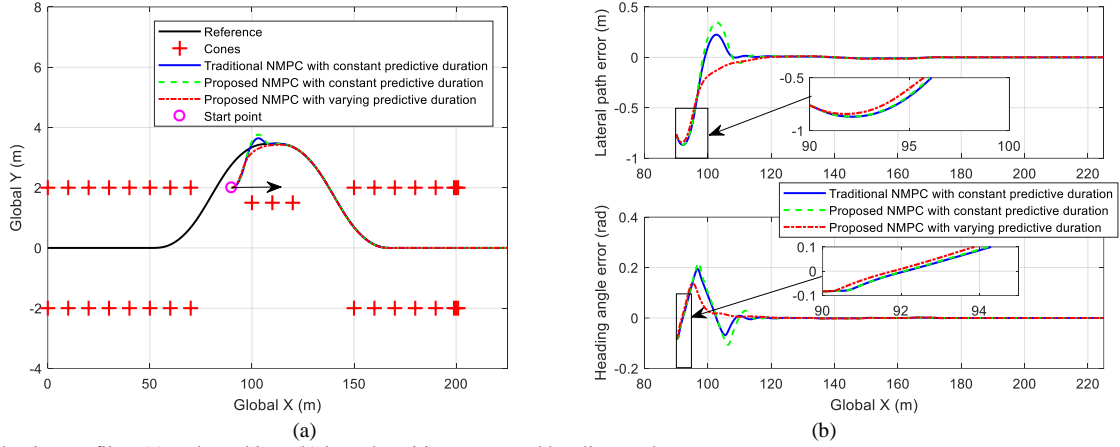


Fig. 7. Path following profiles. (a) path tracking; (b) lateral position errors and heading angle errors.

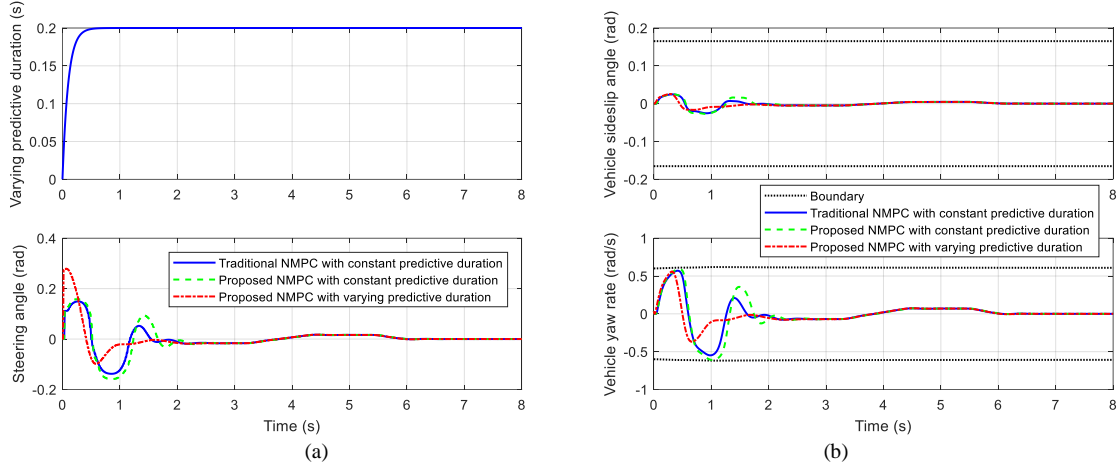


Fig. 8. Yaw motion profiles. (a) varying predictive duration and steering angle; (b) sideslip angle and yaw rate.

### B. Performance Validation of Varying Predictive Duration in Initial Solution Calculation

To verify the performance and calculation efficiency of initial solution optimization, the approaches of varying predictive duration and constant predictive duration are respectively employed in the proposed NMPC controller. Moreover, the traditional NMPC controller in Section IV A is applied as a benchmark here. For fair comparison, three controllers' initial solution is optimized by IP algorithm. The DLC drive cycle with vehicle longitudinal velocity of 50 km/h and road adhesion coefficient of 0.85 is implemented, while the initial position location and vehicle heading angle are [90, 2] m and 0 deg, respectively. That is, there are the initial lateral position error of -0.7614 m and heading error of -0.08348 deg for this path following case.

In this validation, since only one initial solution of varying predictive duration should be optimized in IP algorithm, the calculation time is 0.014 s, much less than that of 0.47 s for  $N_p$  initial solution in constant predictive duration. Focusing on the sample step of 0.02 s in this paper, it signifies the effectiveness of varying predictive duration in fast initial solution optimization.

Fig. 7 shows the path following profiles. From Fig. 7 (a), three controllers can make the drive trajectory convergent to target path at about global position X of 115 m, while that with varying predictive duration illustrates smoother transient

performance and close adjustment time. Fig. 8 demonstrates the varying predictive duration, steering angle, and vehicle yaw motion performance. From the bottom subfigure of Fig. 8 (a), one can observe that by the varying predictive duration, the steering angle command exhibits faster control response at the beginning and then inclines to be smoother, which is attributed to the gradually increasing preview sight in NMPC controller. At the time point of departure, the controller only focuses on a relatively short future target path and hence contributes to reduce the tracking errors as soon as possible. Then at around global position X of 100 m, the predictive horizon increases, and the superior transient response is conducted owing to the inherent advantages of NMPC, as shown in Fig. 7 and Fig. 8. With the short preview sight at the beginning, the response speed regarding sideslip angle and yaw rate by varying predictive duration is faster than that by constant one, as shown in Fig. 8 (b). More intuitive vehicle stability profiles of varying predictive duration are shown in Fig. 9, yielding the small variation range of sideslip angle and sideslip angle rate around original for vehicle stabilization.

It is noteworthy that the greater overshoot effects by constant predictive duration does not conflict to the above explanation that an appropriately longer predictive time size makes more expected transient response. Combining Fig. 7 (b) with Fig. 8, the small control lag by constant predictive duration rises the vehicle yaw rate near its limits at around 1 s. To preferentially

hold the vehicle stability constraints, the NMPC controllers have to weaken its path following effects and ultimately cause the greater overshoot profiles. Hence it can be deduced that under a less extreme cycle, the constant predictive duration approach conducts more desirable performance than varying predictive duration. However, our main focus here is to verify the effectiveness for the calculation efficiency of initial solution optimization and the control effects by varying predictive duration.

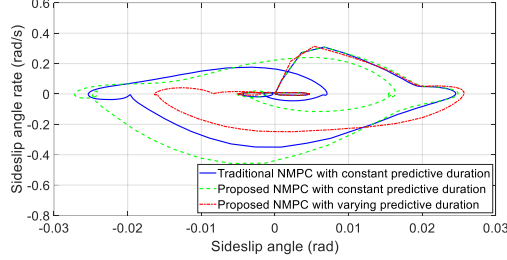


Fig. 9. Phase plane of sideslip angle and sideslip angle rate.

### C. Computational Efficiency Validation

To illustrate the computational efficiency of proposed strategy, the AS and IP algorithms are respectively adopted into NMPC controller for comparison. They are achieved by Matlab® library function “*fmincon*” of the toleration error threshold of 0.01.

Fig. 10 demonstrates the simulation time and execution time which are the given total drive duration and the algorithm run time in real world, respectively. It is noteworthy that the execution time in Fig. 10 is gained by averaging the results from ten runs as to more credibly support our findings. The proposed

C/GMRES algorithm takes only 1.9583 s and 2.998 s, much less than the given simulation time, to complete the path following mission under Case 1 and Case 2, respectively. On the contrary, the execution time of AS and IP algorithms is much greater than the simulation time and even more than 10 times and 46 times longer than those by C/GMRES algorithm, respectively. TABLE III lists the mean, mean square error (MSE), and maximum, for computational time per sample by three algorithms. Compared with AS and IP algorithms, the C/GMRES algorithm implements one to three orders of magnitude reduction under three calculation burden indexes. Its maximum values per sample are only 0.0105 s and 0.0112 s under two drive cases, respectively, signifying that the real-time optimization is realizable by the proposed NMPC controller. Moreover, the MSEs of C/GMRES algorithm are far smaller than those by other two methods, manifesting the more stationary computation time of each optimization. As a whole, it can be believed that the proposed strategy credibly provides the real-time calculation potential for path following application of AEVs in practice.

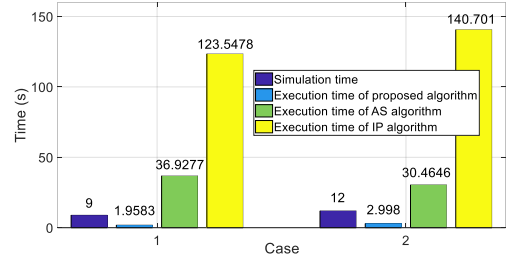


Fig. 10. Simulation time and execution time.

TABLE III. COMPARISON OF COMPUTATIONAL TIME PER SAMPLE.

Test cycle	Algorithm	Mean		MSE		Maximum	
		Value (s)	Calculation burden	Value (s)	Calculation burden	Value (s)	Calculation burden
Case 1	C/GMRES	0.0042	1	$2.0482 \times 10^{-5}$	1	0.0105	1
	AS	0.0833	19.84	0.0079	385.65	0.1767	16.83
	IP	0.2726	64.93	0.0896	4372.37	0.6722	64.02
Case 2	C/GMRES	0.0052	1	$3.0657 \times 10^{-5}$	1	0.0112	1
	AS	0.0487	9.43	0.0029	95.72	0.1724	15.39
	IP	0.2217	42.91	0.0641	2090.89	0.6032	53.86

## V. CONCLUSION

In this paper, a path following control strategy of AEVs is proposed for improving tracking effects and yaw motion stabilization. The NMPC controller is developed to produce the expected front wheels' steering angle command. The C/GMRES algorithm is applied to solving the optimization in NMPC with fast computational efficiency, and the deadzone penalty function is employed to simultaneously handle the inequality constraints and hold the solution's smoothness. In addition, the varying predictive duration is introduced so that the good initial solution can be fast gained by numerical algorithms. The simulation results demonstrate:

1) The proposed strategy is capable of simultaneously conducting superior path following effects and vehicle stabilization. It yields more expected control performance than LQR controller but conducts the similar effects with traditional NMPC controller;

2) The deadzone penalty function is effective for restricting the inequality constraints in C/GMRES algorithm;

3) With the varying predictive duration, the good initial solution can be optimized by numerical algorithms in a computational efficient way. Although its control action will be more radical than the traditional constant predictive duration at the beginning, the overall control performance is acceptable.

4) Compared with AS and IP algorithms, the C/GMRES algorithm can greatly improve the calculation efficiency, manifesting the potential for real-world vehicle application.

Future works will focus on the validation with respect to the control efficacy and the calculation capacity on real-world AEVs. Moreover, the drive cases of variable speed, the improved robustness of NMPC controller, and the modifications regarding constraints handling and varying predictive duration, are arranged as the future study directions.

## REFERENCES

- [1] C. Wei, R. Romano, N. Merat, F. Hajiseyedjavadi, A. Solernou, E. Paschalidis, *et al.*, "Achieving Driving Comfort of AVs by Combined Longitudinal and Lateral Motion Control," in *Advances in Dynamics of Vehicles on Roads and Tracks*, Cham, 2020, pp. 1107-1113.
- [2] K. P. Divakarla, A. Emadi, and S. Razavi, "A Cognitive Advanced Driver Assistance Systems Architecture for Autonomous-Capable Electrified Vehicles," *IEEE Transactions on Transportation Electrification*, vol. 5, pp. 48-58, 2019.
- [3] C. Wei, R. Romano, N. Merat, Y. Wang, C. Hu, H. Taghavifar, *et al.*, "Risk-based autonomous vehicle motion control with considering human driver's behaviour," *Transportation Research Part C: Emerging Technologies*, vol. 107, pp. 1-14, 2019.
- [4] C. Wei, O. A. Olatunbosun, and X. Yang, "A finite-element-based approach to characterising FTire model for extended range of operation conditions," *Vehicle System Dynamics*, vol. 55, pp. 295-312, 2017/03/04 2017.
- [5] E. Ono, S. Hosoe, H. D. Tuan, and S. i. Doi, "Bifurcation in vehicle dynamics and robust front wheel steering control," *IEEE transactions on control systems technology*, vol. 6, pp. 412-420, 1998.
- [6] T. Chen, L. Chen, X. Xu, Y. Cai, and X. Sun, "Simultaneous path following and lateral stability control of 4WD-4WS autonomous electric vehicles with actuator saturation," *Advances in Engineering Software*, vol. 128, pp. 46-54, 2019.
- [7] H. Taghavifar and S. Rakheja, "Path-tracking of autonomous vehicles using a novel adaptive robust exponential-like-sliding-mode fuzzy type-2 neural network controller," *Mechanical Systems and Signal Processing*, vol. 130, pp. 41-55, 2019/09/01/ 2019.
- [8] C. Hu, R. Wang, and F. Yan, "Integral Sliding Mode-based Composite Nonlinear Feedback Control for Path Following of Four-Wheel Independently Actuated Autonomous Electric Vehicles," *IEEE Transactions on Transportation Electrification*, vol. 2, pp. 221-230, 2016.
- [9] C. Hu, Z. Wang, H. Taghavifar, J. Na, Y. Qin, J. Guo, *et al.*, "MME-EKF-based path-tracking control of autonomous vehicles considering input saturation," *IEEE Transactions on Vehicular Technology*, vol. 68, pp. 5246-5259, 2019.
- [10] C. Hu, R. Wang, F. Yan, and H. R. Karimi, "Robust composite nonlinear feedback path-following control for independently actuated autonomous vehicles with differential steering," *IEEE Transactions on Transportation Electrification*, vol. 2, pp. 312-321, 2016.
- [11] K. Bernthorp, R. Quirynen, T. Uno, and S. Di Cairano, "Trajectory tracking for autonomous vehicles on varying road surfaces by friction-adaptive nonlinear model predictive control," *Vehicle System Dynamics*, vol. 58, pp. 705-725, 2020/05/03 2020.
- [12] S. Wei, Y. Zou, X. Zhang, T. Zhang, and X. Li, "An Integrated Longitudinal and Lateral Vehicle Following Control System With Radar and Vehicle-to-Vehicle Communication," *IEEE Transactions on Vehicular Technology*, vol. 68, pp. 1116-1127, 2019.
- [13] Y. Zhang, A. Khajepour, E. Hashemi, Y. Qin, and Y. Huang, "Reconfigurable Model Predictive Control for Articulated Vehicle Stability With Experimental Validation," *IEEE Transactions on Transportation Electrification*, vol. 6, pp. 308-317, 2020.
- [14] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive Active Steering Control for Autonomous Vehicle Systems," *IEEE Transactions on Control Systems Technology*, vol. 15, pp. 566-580, 2007.
- [15] X. Du, K. K. K. Htet, and K. K. Tan, "Development of a Genetic-Algorithm-Based Nonlinear Model Predictive Control Scheme on Velocity and Steering of Autonomous Vehicles," *IEEE Transactions on Industrial Electronics*, vol. 63, pp. 6970-6977, 2016.
- [16] H. Guo, F. Liu, F. Xu, H. Chen, D. Cao, and Y. Ji, "Nonlinear Model Predictive Lateral Stability Control of Active Chassis for Intelligent Vehicles and Its FPGA Implementation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, pp. 2-13, 2019.
- [17] D. Tavernini, M. Metzler, P. Gruber, and A. Sorniotti, "Explicit Nonlinear Model Predictive Control for Electric Vehicle Traction Control," *IEEE Transactions on Control Systems Technology*, vol. 27, pp. 1438-1451, 2019.
- [18] C. Hu, Z. Wang, Y. Qin, Y. Huang, J. Wang, and R. Wang, "Lane keeping control of autonomous vehicles with prescribed performance considering the rollover prevention and input saturation," *IEEE Transactions on Intelligent Transportation Systems*, 2019.
- [19] H. Zhang, G. Zhang, and J. Wang, "Sideslip Angle Estimation of an Electric Ground Vehicle via Finite-Frequency Hinf Approach," *IEEE Transactions on Transportation Electrification*, vol. 2, pp. 200-209, 2016.
- [20] Y. Zou, N. Guo, and X. Zhang, "An integrated control strategy of path following and lateral motion stabilization for autonomous distributed drive electric vehicles," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, p. 0954407019884168, 2019.
- [21] J. Guo, Y. Luo, K. Li, and Y. Dai, "Coordinated path-following and direct yaw-moment control of autonomous electric vehicles with sideslip angle estimation," *Mechanical Systems & Signal Processing*, vol. 105, pp. 183-199, 2018.
- [22] H. B. Pacejka, *Tire and Vehicle Dynamics (Third Edition)*. Oxford: Butterworth-Heinemann, 2012.
- [23] Z. Li, G. Feng, C. Lai, D. Banerjee, W. Li, and N. C. Kar, "Current Injection-Based Multi-parameter Estimation for Dual Three-Phase IPMSM Considering VSI Nonlinearity," *IEEE Transactions on Transportation Electrification*, vol. 5, pp. 405-415, 2019.
- [24] N. Guo, B. Lenzo, X. Zhang, Y. Zou, R. Zhai, and T. Zhang, "A Real-time Nonlinear Model Predictive Controller for Yaw Motion Optimization of Distributed Drive Electric Vehicles," *IEEE Transactions on Vehicular Technology*, 2020.
- [25] A. Gray, M. Ali, Y. Gao, J. K. Hedrick, and F. Borrelli, "A Unified Approach to Threat Assessment and Control for Automotive Active Safety," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, pp. 1490-1499, 2013.
- [26] C. Hu, R. Wang, F. Yan, and C. Nan, "Should the Desired Heading in Path Following of Autonomous Vehicles be the Tangent Direction of the Desired Path?," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, pp. 3084-3094, 2015.
- [27] C. Shen, Y. Shi, and B. Buckham, "Trajectory Tracking Control of an Autonomous Underwater Vehicle Using Lyapunov-Based Model Predictive Control," *IEEE Transactions on Industrial Electronics*, vol. 65, pp. 5796-5805, 2018.
- [28] H. K. Khalil, *Nonlinear Systems*: Prentice Hall, 2002.
- [29] J. V. Frasch, A. J. Gray, M. Zanon, H. J. Ferreau, and M. Diehl, "An auto-generated nonlinear MPC algorithm for real-time obstacle avoidance of ground vehicles," in *Control Conference*, 2013.
- [30] D. E. Kirk, "Optimal Control Theory: An Introduction," *Dover Publications*, 2004-04-30.
- [31] E. L. Allgower and K. Georg, "Numerical Continuation Methods: An Introduction," *Mathematics of Computation*, vol. 13, pp. xxvi,388, 2003.
- [32] D. A. Knoll and D. E. Keyes, "Jacobian-free Newton-Krylov methods: a survey of approaches and applications," *Journal of Computational Physics*, vol. 193, pp. 357-397, 2004/01/20/ 2004.
- [33] U. Rosolia, S. D. Bruyne, and A. G. Alleyne, "Autonomous Vehicle Control: A Nonconvex Approach for Obstacle Avoidance," *IEEE Transactions on Control Systems Technology*, vol. 25, pp. 469-484, 2017.
- [34] J. M. Maciejowski, "Predictive Control with Constraints," *London, UK : Pearson Education*, 2002.
- [35] T. Ohtsuka, "A continuation/GMRES method for fast computation of nonlinear receding horizon control," *Automatica*, vol. 40, pp. 563-574, 2004.
- [36] S. A. Sajadi-Alamdari, H. Voos, and M. Darouach, "Nonlinear Model Predictive Control for Ecological Driver Assistance Systems in Electric Vehicles," *Robotics and Autonomous Systems*, 2019.
- [37] C. Dugas, Y. Bengio, F. B  lisle, C. Nadeau, and R. Garcia, "Incorporating second-order functional knowledge for better option pricing," in *Advances in neural information processing systems*, 2001, pp. 472-478.
- [38] M. Huang, H. Nakada, K. Butts, and I. Kolmanovsky, "Nonlinear Model Predictive Control of a Diesel Engine Air Path: A Comparison of Constraint Handling and Computational Strategies," *Ifac Papersonline*, vol. 48, pp. 372-379, 2015.
- [39] S. Chao, B. Buckham, and S. Yang, "Modified C/GMRES Algorithm for Fast Nonlinear Model Predictive Tracking Control of AUVs," *IEEE Transactions on Control Systems Technology*, vol. PP, pp. 1-9, 2016.
- [40] M. S. Corporation. (2019). *CarSim Information* [Online]. Available: <https://www.carsim.com/products/carsim/index.php>
- [41] C. Hu, Y. Qin, H. Cao, X. Song, K. Jiang, J. J. Rath, *et al.*, "Lane keeping of autonomous vehicles based on differential steering with adaptive multivariable super-twisting control," *Mechanical Systems and Signal Processing*, vol. 125, pp. 330-346, 2019.
- [42] A. E. Bryson, *Applied optimal control: optimization, estimation and control*: Routledge, 2018.